Interfacing with the Raspberry Pi Pico

# VsCode

## Extensions

Install the **MicroPico** and **Raspberyy Pi Pico** extensions:

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

## References:

[SDK-Setup](https://www.raspberrypi.com/documentation/microcontrollers/c_sdk.html#sdk-setup)

[Getting Started with Pico(PDF)](https://datasheets.raspberrypi.com/pico/getting-started-with-pico.pdf)

## Loading and Running Code

### VsCode: C/C++

1. Hold down the BOOTSEL button on your Pico-series device while plugging it into your development device using a micro USB cable to force it into USB Mass Storage Mode.
   1. File explorer will pop up:
   2. A screenshot of a computer

      Description automatically generated
2. Click on the **Raspberry Pi Pico Project** extension button

A screenshot of a computer program

Description automatically generated

1. Click on the **Project** -> **Compile Project** button
2. Click on the **Project** -> **Run Project (USB)** button
3. You should see the terminal tab at the bottom of the window open. It will display information concerning the upload of the code. Once the code uploads, the device will reboot. Your file explorer window will also automatically close, and your program will immediately start running.

A screen shot of a computer

Description automatically generated

1. If serial monitor is enabled in your code you can see output by connecting with the following options, this can be done via putty or the Serial Monitor windows in VsCode:
   1. Serial Line: **COM<Port>**
   2. Baud Rate: **115200**

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

### Drag and Dropping: C/C++

You can also load code by drag and dropping

1. Hold down the BOOTSEL button to make the Raspberry Pi Pico file explorer window pop up like in the previous step.
2. Compile the code like in the previous step.
3. This will generate a .uf2 file. Drag and drop this into the file explorer window.

A screenshot of a computer

Description automatically generated

1. The file explorer window will automatically close, and the code will immediately start running.

### VsCode: MicroPython

1. If you previously loaded a C/C++ program into the Pico and now have a MicroPython project open, upon connecting it you will see the following error:

A screenshot of a computer error

Description automatically generated

1. Unplug the Pico, and hold down the BOOTSEL button on your Pico-series device while plugging it into your development device using a micro USB cable to force it into USB Mass Storage Mode.
2. Now you will see the following message. Click on Yes for the first pop up, and Yes or No depending on if you have a Wireless board
   1. **TODO: I actually don’t know what the second pop up is talking about**

A screenshot of a computer

Description automatically generated

A screen shot of a phone

Description automatically generated

1. Once flashing is successful you will see the following:

A black and white text

Description automatically generated

1. Now you can write your MicroPython code, right click on the file and either **Upload file to Pico** or **Run current file on Pico**:
   1. **Note:** When testing name your files something other than **main.py**. Only name your file main.py if you want your script to run automatically upon startup (e.g. your final code). When the Pico boots, it searches for a file named main.py in its file system and executes it. If this file contains infinite loops without interrupt handling, the Pico may become unresponsive and seem bricked. See the **Common Errors** section.
   2. **Note:** The Upload option does not seem to work. It might be that it can only be uploaded if its named main.py

A screenshot of a computer program

Description automatically generated

### Thony: MicroPython

1. Open Thony
2. Hold down the BOOTSEL button on your Pico-series device while plugging it into your development device using a micro USB cable to force it into USB Mass Storage Mode.
3. Click-on the bottom right corner and select **Install MicroPython**:

A screenshot of a computer program

Description automatically generated

1. Choose the following options then **Install**:

A screenshot of a computer

Description automatically generated

1. Once finished, you should see **Done!** At the bottom left corner of the previous window. Now **Close**.
2. Click-on the bottom right corner and select the right board:

A screenshot of a computer program

Description automatically generated

1. Now you can start running your code on the Pico by clicking the run button.
2. If a file is already saved in the Pico (e.g. ledBlink.py) you can invoke it via the **Read-Eval-Print Loop (REPL)** (an interactive programming environment) as follows:

A screenshot of a computer error message

Description automatically generated

* 1. Press Ctrl + C to exit

1. This means you can have more than one file saved into the Pico that you can invoke

### Common Errors

If the board appears bricked and you can’t connect to it even after flashing the firmware (maybe because you wrote a file named main.py that cannot exit) follow the steps [HERE](https://forums.raspberrypi.com/viewtopic.php?f=146&t=305432H). This FW renames the main.py to something else or removes it.